Technical Cybersecurity

ROP Concepts

History

Successor to ret2libc attacks

- Ret2libc from W^X stacks
- ROP from implementation of ASLR

First released in 2007

- Hovav Shacham at UCSD
- Turing-complete

JOP was the next new attack, 2011

- Tyler Bletsch, et. al. NCSU and National Univ. Singapore

ROP

Return-oriented programming

- A way to write new programs with the programs own code
- Works because:
  - We can read from any start address
  - x86 code is not bound to specific word lengths
  - x86 code is “dense” - a little code can do quite a bit
- Similar to Jump-oriented programming
  - ROP uses ret instructions
  - JOP uses jmp {register} instructions

Attack Design

Multistaged attacks

- ROP used to turn off security controls
- Then a second payload exploits the program
  - e.g. use ROP to turn off ASLR

Single payload attacks

- ROP is Turing complete
- Can execute arbitrary programs
  - Though you and I probably don't have the patience to do this :-)

Uses Particular Structure

Opcode followed by RET

- e.g. pop eax; ret

Or a sequence

- e.g. pop eax; pop edx; ret

The RET is common

- Detection techniques monitor for frequent RET instructions

Enter JOP and Others

RET REPLACEMENT

- JOP
- x86: jmp and pop instructions (e.g. pop the return address from the stack into EIP or pop into a different register and jump from there)
- ARM: load and branch instructions (e.g. load an an address into a register and use a branch instruction on the register address)

POSSIBLE TO STILL DETECT

- Look for frequent JMPs
- Not commonly implemented

How do you protect?